Incremental Learning of Probabilistic Rules from Clinical Databases based on Rough Set Theory

Shusaku Tsumoto and Hiroshi Tanaka
Department of Information Medicine
Medical Research Institute, Tokyo Medical and Dental University
1-5-45 Yushima, Bunkyo-ku Tokyo 113 Japan
E-mails:tsumoto.com@mri.tmd.ac.jp, tanaka@cim.tmd.ac.jp

Several rule induction methods have been introduced in order to discover meaningful knowledge from databases, including medical domain. However, most of the approaches induce rules from all the data in databases and cannot induce incrementally when new samples are derived. In this paper, a new approach to knowledge acquisition, which induce probabilistic rules incrementally by using rough set technique, is introduced and was evaluated on two clinical databases. The results show that this method induces the same rules as those induced by ordinary non-incremental learning methods, which extract rules from all the datasets, but that the former method requires more computational resources than the latter approach.

INTRODUCTION

Several symbolic inductive learning methods, such as induction of decision trees[1,5], and AQ family[3], have been proposed to discover meaningful knowledge from large databases. However, most of the approaches induce rules from all the data in databases, and cannot induce incrementally when new samples are derived. So, we have to apply rule induction methods again to the databases when such new samples are given, which causes the computational complexity to be expensive even if the complexity is $O(n^2)$.

Therefore, it is important to develop incremental learning systems in order to manage large databases[6,8]. However, most of the previously introduced learning systems have the following two problems: first, those systems do not outperform ordinary learning systems, such as AQ15[3], C4.5[5] and CN2[2]. Secondly, those incremental learning systems mainly induce deterministic rules, which are often overfitted to datasets. Thus, it is indispensable to develop incremental learning systems which induce probabilistic rules to solve the above two problems.

Extending concepts of rule induction methods based on rough set theory, we introduce a new approach to knowledge acquisition, which induces probabilistic rules incrementally, called PRIMEROSE-INC (Probabilistic Rule Induction Method based on Rough Sets for Incremental Learning Methods).

Although the previously introduced rule induction method PRIMEROSE[7], which extracts rules from all the data in database uses apparent accuracy to search for probabilistic rules, PRIMEROSE-INC first uses coverage to search for the candidates of rules, and secondly uses accuracy to select from the candidates.

This system was evaluated on two clinical databases: databases on meningoencephalitis and databases on headache with respect to the following four points: accuracy of classification, the number of generated rules, spatial computational complexity, and temporal computational complexity. The results show that PRIMEROSE-INC induces the same rules as those induced by PRIMEROSE, but that the former method requires more computational resources than the latter approach.

PROBABILISTIC RULES

Rough Set Theory

Rough set theory clarifies set-theoretic characteristics of the classes over combinatorial patterns of the attributes, which are precisely discussed by Pawlak[4]. This theory can be used to acquire some sets of attributes for classification and to evaluate how precisely the attributes are able to classify data.

Table 1: An Example of Database

No.	loc	nat	his	nau	class
1	who	per	per	no	m.c.h.
2	who	per	per	no	m.c.h.
3	lat	thr	per	no	migraine
4	who	thr	per	yes	migraine
5	who	per	per	no	psycho

NOTATIONS: loc: location, nat: nature, his: history, nau: nausea,
who: whole, lat: lateral, per: persistent,
thr: throbbing, m.c.h.: muscle contraction headache,
migraine: classic migraine,
psycho: psychogenic headache.

Let us illustrate the main concepts of rough sets which are needed for our formulation. Table 1 is a small example of database which collects the patients who complain of headache. First, let us consider how an attribute "loc" classify the headache patients' set of the table. The set whose member satifies [loc=who] is {1,2,4,5}, which shows that the attribute "loc" of 1st, 2nd, 4th and 5th cases is equal to "who"(In the following, the numbers in a set are used to represent each record number). This set means that we cannot classify {1,2,4,5} further solely by using the constraint R=[loc=who]. This set is defined as the indiscernible set over the relation R and described as follows: $[x]_R = \{1,2,4,5\}$. In this set, {1,2} suffer from muscle contraction headache("m.c.h."), {4} from classical migraine("migraine"), and {5} from psycho("psycho"). Hence we need other additional attributes to discriminate between "m.c.h.", "migraine", and "psycho". Using this concept, we can evaluate the classification power of each attribute. For example, "nat=thr" is specific to the case of classic migraine("migraine"). We can also extend this indiscernible relation to multivariate cases, such as $[x]_{[loc=who]\wedge[nau=no]} = \{1,2\}$ and $[x]_{[loc=who]\vee[nat=no]} = \{1,2,4,5\}$, where ^ and v

1091—8280/97/$5.00 © 1997 AMIA, Inc.

denote "and" and "or" respectively. In the framework of rough set theory, the set {1,2} is called *strictly definable* by the former conjunction, and also called *roughly definable* by the latter disjunctive formula.

In this way, the classification of training samples *D* is characterized in the set-theoretic framework:Steps for classification are defined as search methods for the best set $[x]_R$ which is supported by the relation *R*. Moreover, two important statistical measures for classification, accuracy and coverage (true positive rate) are defined as:

$$\alpha_R(D) = |[x]_R \mathrm{I} \ D|/|[x]_R|, \kappa_R(D) = |[x]_R \mathrm{I} \ D|/|D|$$

where |A|, $\alpha_R(D)$, and $\kappa_R(D)$ denote the cardinality of a set A, an accuracy of R as to classification of D, and a coverage, or a true positive rate of R to D, respectively. For example, when R and D are set to [nau=yes] and [class=migraine], $\alpha_R(D)$ =1/1=1.0 and $\kappa_R(D)$ =1/2=0.50.

For further information on rough set theory, readers could refer to [4,9].

Definition of Probabilistic Rules

The most simplest probabilistic rules are classification rules which have high accuracy and high coverage.[1] Such rules can be defined as:

$$R \longrightarrow d, R = \wedge[ai = vj], \alpha_R(D) > \delta_\alpha, \kappa_R(D) > \delta_\kappa$$

where *d* denote a class to which all the members of *D* belong and where δ_α and δ_κ denote given thresholds for accuracy and coverage, respectively. For the above example shown in Table 1, probabilistic rules for m.c.h. are given as follows:

[loc=who]&[nau=no] → m.c.h. α=2/3=0.67, κ=1.0,
[nat=per] → m.c.h. α=2/3=0.67, κ=1.0.

where δ_α and δ_κ are set to 0.5 and 0.3, respectively. It is notable that
this rule is a kind of probabilistic proposition with two statistical measures, which is one kind of an extension of Ziarko's variable precision model(VPRS)[9].[2]

PROBLEMS IN INCREMENTAL RULE INDUCTION

The most important problem in incremental learning is that, even in an applied domain where the rules are deterministic, it does not always induce the same rules as those induced by ordinary learning systems[8].[3] Furthermore, since induced results are strongly dependent on the former training samples, the tendency of overfitting is larger than the ordinary learning systems.

The most important factor of this tendency is that the revision of rules is based on the formerly induced rules, which is the best way to suppress the exhaustive use of computational resources. However, when induction of the same rules as ordinary learning methods is required, more computational resources will be needed, because all the candidates of rules should be considered.

Thus, for each step, computational space for deletion of candidates and addition of candidates is needed, which causes the computational speed of incremental learning to be slow. Moreover, in case when probabilistic rules should be induced, the situation becomes much severer, since the candidates for probabilistic rules become much larger than those for deterministic rules.

In our approach, we first focus on the performance of incremental learning methods, that is, we introduce a method which induces the same rules as those derived by ordinary learning methods. Then, we estimate the effect of this induction on computational complexity.

AN ALGORITHM

In order to provide the same classificatory power to incremental learning methods as ordinary learning algorithms, we introduce
an incremental learning method PRIMEROSE-INC(Probabilistic Rule Induction Method based on Rough Sets for Incremental Learning Methods). PRIMEROSE-INC first measures the statistical characteristics of coverage of elementary attribute-value pairs. There it measures the statistical characteristics of accuracy of the whole pattern of attribute-value pairs observed in a dataset.

In this algorithm, we use the following characteristic of coverage.

Proposition 1 (Monotonicity of Coverage)

Let R_{i+1} denote an attribute-value pair, which is a conjunction of R_i and $[a_{i+1}=v_{i+1}]$. Then,

$$\kappa_{Ri+1}(D) \leq \kappa_{Ri}(D).$$

Proof. Since $[x]_{Ri+1} \subseteq [x]_{Ri}$ holds,

$$\kappa_{Rj}(D) = |[x]_{Ri+1} \cap D|/|D| \leq |[x]_{Ri} \cap D|/|D| = \kappa_{Ri}(D).$$

Furthermore, in rule induction methods, R_{i+1} is selected to satisfy $\alpha_{Ri+1}(D) \geq \alpha_{Ri}(D)$. Therefore, it is sufficient to check the behavior of coverage of elementary attribute-value pairs in order to estimate the characteristics of induced rules, while it is necessary to check the behavior of accuracy of elementary attribute-value pairs and accuracy of patterns observed in the databases in order to estimate the characteristics of induced rules.

Algorithm for Rule Induction. From these considerations, the

[1] In this model, we assume that accuracy is dominant over coverage.

[2] In VPRS model, the two precisions of accuracy is given, and the probabilistic proposition with accuracy and two precision conserves the characteristics of the ordinary proposition. Thus, our model is to introduce the probabilistic proposition not only with accuracy, but also with coverage.

[3] Here, ordinary learning systems denote methods that induces all rules by using all the samples.

selection algorithm is defined as follows, where the following four lists are used. $List_1$ and $List_2$ store elementary relations which decrease and increase its coverage, respectively, when a new training sample is given. $List_a$ is a list of probabilistic rules which satisfies the condition on the thresholds of accuracy and coverage. Finally, $List_r$ stores a list of probabilistic rules which do not satisfy the above condition.

For rule induction, the following steps are applied to each class *d*, the set of which is equal to *D*.

(1) Revise a coverage and an accuracy of each elementary attribute value pair $[a_i=v_j]$ by using a new additional sample S_k.
(2) For each pair $r_{ij}=[a_i=v_j]$, κ_{rij}(D) decreases, then store it into $List_1$. Else, store it into $List_2$.
(3) For each member r_{ij} in $List_1$, search for a rule in $List_a$ whose condition *R* includes r_{ij} and which fails to satisfies $\alpha_{Ri}(D) \geq \delta_\alpha$ and $\kappa_{Ri}(D) \geq \delta_\kappa$. Remove it from $List_a$, and Store it into $List_r$.
(4) For each member r_{ij} in $List_2$, search for a rule in $List_r$, whose condition *R* includes r_{ij}. If it satisfies $\alpha_{Ri}(D) \geq \delta_\alpha$ and $\kappa_{Ri}(D) \geq \delta_\kappa$, then remove it from $List_r$ and store it into $List_a$. Otherwise, search for a rule which satisfies the above condition by using rule induction methods.[4]
(5) For other rules in $List_r$, revise accuracy and coverage. If a rule satisfies $\alpha_{Ri}(D) \geq \delta_\alpha$ and $\kappa_{Ri}(D) \geq \delta_\kappa$, then remove it from $List_r$ and store it into $List_a$.

For example, let us consider a case when the following new sample is provided after all the probabilistic rules are induced from Table 1:[5]

No.	loc	nat	his	nau	class
6	lat	thr	per	no	m.c.h.

The initial condition of this system derived by Table 1 is summarized into Table 2, and $List_a$ and $List_r$ for m.c.h. are given as follows: $List_a$={[loc=who]&[nau=no],[nat=per]},and
$List_r$={[loc=who],[loc=lat],[nat=thr],[his=per],[nau=yes],
[nau=no] }.

Table 2: Accuracy and Coverage of Elementary Relations (m.c.h.)

Relation	Accuracy	Coverage
[loc=who]	0.5	1.0
[loc=lat]	0.0	0.0
[nat=per]	0.67	1.0
[nat=thr]	0.0	0.0
[his=per]	0.4	1.0
[nau=yes]	0.0	0.0
[nau=no]	0.5	1.0

Then, the first step revises accuracy and coverage for all the elementary relations (Table 3). Since the coverages of [loc=lat], [nat=thr], and [nau=yes] become larger than 0.3, they are included in $List_2$. In the same way, [loc=who], [nat=per], and [nau=no] are included in $List_1$.

Table 3: Revised Accuracy and Coverage of Elementary Relations (m.c.h.)

Relation	Accuracy	Coverage
[loc=who]	0.5	**0.67**
[loc=lat]	**0.5**	**0.33**
[nat=per]	0.67	**0.67**
[nat=thr]	**0.33**	**0.33**
[his=per]	**0.5**	1.0
[nau=yes]	**0.5**	**0.33**
[nau=no]	**0.4**	**0.67**

Next, the third step revises two measures for all the rules in $List_a$ whose conditional parts include a member of $List_1$ Then, the formerly induced probabilistic rules are revised into:

[loc=who]&[nau=no] → m.c.h. α=0.67, κ=0.67,
[nat=per] → m.c.h. α=0.67, κ=0.67.

and not one of them is removed from $List_a$.

Then, the fourth step revises two measures for all the rules in $List_r$ whose conditional parts include a member of $List_2$. Then, the following probabilistic rule satisfies α>0.5 and κ>0.3:

[loc=lat]&[nau=yes] → m.c.h. α=1.0, κ=0.33,

and is stored into $List_a$. Finally, $List_a$ and $List_r$ for m.c.h. are calculated as follows:
$List_a$={[loc=who]&[nau=no], [nat=per]&[nau=no],
[loc=lat]&[nau=yes]}, and
$List_r$={[loc=who], [loc=lat], [nat=per], [nat=thr], [his=per],
[nau=yes], [nau=no] }.

EXPERIMENTAL RESULTS

PRIMEROSE-INC was evaluated to the following two clinical databases and compared with CN2[2], AQ15[3], C4.5[5], and PRIMEROSE[7]. One is on differential diagnosis of headache, which consists of 1477 samples, 10 classes, and 20 attributes. The other one is on differenital diagnosis of meningitis, which consists of 198 samples, 3 classes, and 25 attributes.

The experiments were performed by the following three steps.
First, these samples are randomly splits into pseudo-training samples and pseudo-test samples. Second, by using the pseudo-training

[4] That is, it makes a conjunction of attribute-value pairs and checks whether this conjunction satisfies the condition on a probabilistic rule: $\alpha_{Ri}(D) \geq \delta_a$ and $\kappa_{Ri}(D) \geq \delta_k$.

[5] In this example, thresholds for accuracy and coverage, δ_a and δ_k are again set to 0.5 and 0.3, respectively.

samples, PRIMEROSE-INC and other four systems induces rules and the statistical measures.[6] Third, the induced results are tested by the pseudo-test samples. These procedures are repeated for 100 times and average the estimators for accuracy of diagnosis over 100 trials.

Table 4: Experimental Results: Accuracy and Number of Rules (Headache)

Method	Accuracy	Number of Rules
PRIMEROSE-INC	89.5±5.4%	67.3±3.0
PRIMEROSE	89.5±5.4%	67.3±3.0
C4.5	85.8±2.4%	16.3±2.1
CN2	87.0±3.9%	19.2±1.7
AQ15	86.2±2.6%	31.2±2.1

Table 5: Experimental Results: Accuracy and Number of Rules (Meningitis)

Method	Accuracy	Number of Rules
PRIMEROSE-INC	81.5±3.2%	52.3±1.4
PRIMEROSE	81.5±3.2%	52.3±1.4
C4.5	74.0±2.1%	11.9±3.7
CN2	75.0±3.9%	33.1±4.1
AQ15	80.7±2.7%	32.5±2.3

Table 4 and 5 give the comparison between PRIMEROSE-INC and other rule induction methods with respect to the averaged classification accuracy and the number of induced rules. These results show that PRIMEROSE-INC attains the same performance of PRIMEROSE, which is the best performance in those rule induction systems.

Table 6: Experimental Results: Spatial and Temporal Complexity (Headache)

Method	Code-Area	Cul-CPU time
PRIMEROSE-INC	18241±219	4027±61
PRIMEROSE	1210±98	107403±219
C4.5	-	79198±193
CN2	-	118197±211
AQ15	-	120192±108

DEFINITION. Cul-CPU time: Cumulative CPU Time

Table 7: Experimental Results: Spatial and Temporal Complexity (Meningitis)

Method	Code-Area	Cul-CPU time
PRIMEROSE-INC	1241±34	1902±710
PRIMEROSE	210±14	16269±508
C4.5	-	11862±707
CN2	-	11117±504
AQ15	-	12117±299

Table 6 and 7 give the comparison between PRIMEROSE-INC and other rule induction methods with respect to the code area,
and cumulative CPU time, which denotes how much time is totally needed to rerun the rule induction methods from scratch when a new sample is added. These results show that PRIMEROSE-INC outperforms all the other non-incremental learning methods, although they need much larger memory space for running. Furthermore, the comparison of PRIMEROSE-INC with PRIMEROSE suggest that the computational resources needed for incremental learning are much larger than those for ordinary learning in order that incremental learning methods induce all the same results as the ordinary learning methods.

DISCUSSION AND RELATED WORK

Applicability to Clinical Practice

As mentioned earlier, one of the important practical limits of non-incremental learning systems is that these methods have to be re-executed when a new additional sample is given, which causes the computational complexity to be expensive even if the complexity is polynominal, as shown in Table 6 and 7. It is one of the reasons why such classification systems have not been used for real-sized applications, such as the analysis of actual computerized patient records, because data acquisition of medical patient records is very dynamic.

On the other hand, one of the practical limits of incremental learning systems is that they suffer from the problem on sampling order: induced results will be different if the order of training samples is changed. Thus, in order to apply incremental learning methods to practical situation, we have to solve the problem on sampling order. The introduced method PRIMEROSE-INC solves this problem, which suggests that the applicability of our approach is much wider than that of other systems.

Proposition Logic and First Order Logic

The introduced method induces probabilistic propositions incrementally from databases. Another possiblity of rule induction is to induce first order-relations from datasets, which have been studied in research on logic programming. This research area is called *inductive logic programming* (ILP), based on the subsumption technique[10]. The advantages of ILP over proposition rule induction methods is that ILP systems use domain knowledge clearly and utilize induced knowledge as new domain knowledge. This usage of induced knowledge can be viewed as incremental learning. However, unfortunately, ILP systems views the incorporation of new domain knowledge as an intermediate step to induce important rules, not as revision of rules. Thus, rule induction systems using ILP are classified into non-incremental learning methods and they suffer from the same problems as non-incremental rule induction methods mentioned earlier.

[6] The thresholds δ_a and δ_k were set to 0.75 and 0.5, respectively in these experiments.

Decision Matrix

Shan and Ziarko introduce decision matrix method, which is based on an indiscernible matrix, in order to make incremental learning methods efficient[6].

Their approach is simple, but very powerful. For the above example shown in Table 1, the decision matrix for m.c.h. is given as Table 8, where the rows denote positive examples of m.c.h., the columns denote negative examples of m.c.h., and each matrix element a_{ij} shows the differences in attribute-value pairs between *i*th sample and *j*th sample. Also, ϕ denotes that all the attribute-value pairs in two samples are the same.

Table 8: Decision Matrix for m.c.h.

U	3	4	5
1	(l=w), (n=p)	(n=p), (n_a=n)	ϕ
2	(l=w), (n=p)	(n=p), (n_a=n)	ϕ

NOTATIONS: l=w: loc=who, n=p: nat=per, n_a=n: nau=no

Shan and Ziarko discuss induction of deterministic rules in their original paper, but it is easy to extend it into probabilistic domain. In Table 8, the appearance of ϕ shows that decision rules for m.c.h. should be probabilistic. Since the first and the second row have the same pattern, {1,2,5} have the same pattern of attribute-value pairs, whose accuracy is equal to 2/3=0.67. Furthermore, rules are obtained as:

$[loc=who]\vee[nat=per])\wedge([nat=per]\vee[nau=no]) \rightarrow$ m.c.h.,
which are exactly the same as shown in Section 3.

When a new example is given, it will be added to a row when it is a positive example, and a column when a negative example. Then, again, new matrix elements will be calculated. For the above example, the new decision matrix will be obtained as in Table 9.

Table 9: Decision Matrix with Additional Sample

U	3	4	5
1	(l=w), (n=p)	(n=p), (n_a=n)	ϕ
2	(l=w), (n=p)	(n=p), (n_a=n)	ϕ
6	(n_a=y)	(l=l)	(l=l), (n=t), (n_a=y)

NOTATIONS: l=w/l: loc=who/lat, n=p/t : nat=per/thr,
n_a=y/n: nau=yes/no

Then, from the last row, the third rule: $[loc=lat]\wedge[nau=yes] \rightarrow$m.c.h. is obtained.

The main difference between our method and decision matrix is that the latter approach is based on apparent accuracy, rather than coverage. While the same results are obtained in the above simple example, the former approach is sensitive to the change of coverage and the latter is to the change of accuracy. Thus, if we need rules of high accuracy, decision matrix technique is very powerful. However, a rule of high accuracy may support only a small case, which suggests that this rule is overfitted to the training samples and coverage should be dominant over accuracy in order to suppress the tendency of overfitting. However, original decision matrix technique does not incorporate such calculation of coverage. Thus, it needs to include such calculation mechanism when we extend it into the usage of both statistical measures.

CONCLUSION

In this paper, a new approach to incremental induction of probabilistic rules, called PRIMEROSE-INC, is introduced, which is based on the extension of variable precision rough set model. This system was evaluated on two clinical databases. The experimental results show that the introduced system induces the same rules as those induced by PRIMEROSE, but that the former method requires much computational resources than the latter approach.

References

1. Breiman, L., Freidman, J., Olshen, R., and Stone, C. Classification And Regression Trees. Belmont, CA: Wadsworth International Group, 1984.
2. Clark, P., Niblett, T. The CN2 Induction Algorithm. Machine Learning, 1989; 3: 261-283.
3. Michalski, R. S., Mozetic, I., Hong, J., and Lavrac, N. The Multi-Purpose Incremental Learning System AQ15 and its Testing Application to Three Medical Domains. Proceedings of the fifth National Conference on Artificial Intelligence, pp. 1041-1045, AAAI Press, Palo Alto, CA, 1986.
4. Pawlak, Z. Rough Sets. Kluwer Academic Publishers, Dordrecht, 1991.
5. Quinlan, J.R. C4.5 - Programs for Machine Learning, Morgan Kaufmann, CA., 1993.
6. Shan, N. and Ziarko, W. Data-Based Acquisition and Incremental Modification of Classfication Rules. Computational Intelligence, 1995; 11: 357-370.
7. Tsumoto, S. and Tanaka, H. PRIMEROSE: Probabilistic Rule Induction Method based on Rough Sets and Resampling Methods. Computational Intelligence, 1995; 11: 389-405.
8. Utgoff, P.E. Incremental Learning of Decision Trees. Machine Learning, 1989; 4: 161-186.
9. Ziarko, W. Variable Precision Rough Set Model. Journal of Computer and System Sciences, 1993; 46: 39-59.
10. Bergadano, F. and Gunetti, D. Inductive Logic Programming, MIT press, 1995.